A Hybrid Deep Neural Network System and its Applications

A synopsis of the proposed thesis to be submitted for the degree of

Doctor of Philosophy

in

Computer Science

submitted by

Soniya

under the supervision of

Dr. Sandeep Paul
Supervisor
Associate Professor
Dept. of Physics & Computer Science

Dr. Lotika Singh
Co-supervisor
Assistant Professor
Dept. of Physics & Computer Science

forwarded by

Prof. G. S. Tyagi
Head
Dept. of Physics & Computer Science

Prof. Ravindra Kumar
Dean
Faculty of Science

Department of Physics & Computer Science
Dayalbagh Educational Institute
(Deemed University)
Dayalbagh, Agra UP 282 005

September, 2015

1 Introduction

Recognizing a beautiful butterfly sitting on a flower in your garden, different shapes of clouds in the sky, a cup of tea on your table, the words on this paper, or many different kind of things that we see in our daily lives seem to be an easy task for us. A human brain does this within a fraction of second even if they are appearing with different poses and in different environment. In this context the obvious question that comes in our mind is that how does the brain recognizes such varied patterns or how does it classify these objects based on their appearance? In finding the answers to some of these questions led to the evolution of the interesting concept of Artificial Neural Networks (ANN's). ANN's derives its inspiration from the biological brain. ANN's provide one of the most powerful classes of solutions for machine learning [1, 2, 3, 4, 5, 6, 7].

In the machine learning community, neural networks techniques are widely used for a variety of problems like pattern recognition, optimization, control systems, classification, etc. Classical neural networks in association with fuzzy set theory forms an excellent combination in handling linguistic and numerical data at the same time. The uncertainty or the imprecise information which occurs in real world data can be effectively handled by integrating fuzzy logic to the classical neural networks. When used together in a cohesive manner they provide us with intelligent systems capable to utilizing benefits of both the technologies [8, 9, 10]. The integration of these two technologies along with evolutionary computation (used mainly for optimization) has been in the forefront of the machine learning community for a long time now [11, 12]. Research in the machine learning community still revolves around making more powerful and intelligent systems by having inspiration from different parts of the human brain [13, 14, 15, 16].

One of the potential problems usually addresses by researches in the field is that of feature selection. Feature extraction plays a vital role in pattern recognition. It heavily effects the performance of network. In feature extraction we extract those aspects of input data which contain more information. One way is to use principle component analysis (PCA) which reduces high dimensional data to low dimensional data by ignoring other irrelevant dimensions [2]. The functioning of PCA is limited by its linear transformation of input from one space m to another space n (such that $n < m$). And thus this technique is unable to handle complex non-linear correlations. As another approach, multilayer perceptron networks can also be used for dimensionality reduction. The network with m inputs, m outputs and n hidden units (such that $n < m$) with

linear activation function maps m dimensional data into n dimensional data. Such networks are known as auto-encoder or auto-associator [2]. This approach is equivalent to PCA because it is also linearly transforming high dimensional data into low dimensional data. The network with more hidden layers and non-linear activation functions can be used to capture more complex correlations. This approach of dimensionality reduction is known as nonlinear generalization of PCA. The learning of such networks with conventional back-propagation algorithm suffers from the problem of vanishing gradients [17].

In another approach, some pre-processing is performed to extract relevant features. The extracted features are then used as input for neural network based classifier [18]. This kind of feature extractor is expensive in terms of inputs because it requires sufficient amount of engineering skills and domain knowledge. Several existing feature selection techniques which are both offline and online include [19, 6, 20] and [21, 22] respectively. In general, classical neural network based classifiers require good feature extractor to meticulously disentangle those characteristics of input that are really essential for classification. More specifically, in case of images and signals some local correlations may be present in the nearby pixels which may get ignored by classical classifier i.e., the topology of an input may not be taken into account for a particular task. These networks may not be effective in solving the selectivity-invariance dilemma [23]. This issue is addressed by deep learning. Deep learning techniques reinforce the discriminative power of simple neural networks. It automatically extracts essential features from raw data and makes them robust with respect to variations in input [24, 25, 26, 23, 27].

2 Deep Neural Networks

Deep neural networks are inspired by functioning of the visual cortex where each layer receive input from a layer below it and transform its representation and then propagate it to the layer above it. The term deep comes from the fact that the depth in network is more as compared to the conventional neural networks. Because of this, sometimes conventional neural networks are also called as shallow networks. A simple network with one hidden layer is sufficient to approximate any continuous function. By introducing one more hidden layer we can perform the task efficiently. It is because the first hidden layer extracts local features and the second layer extracts global features from the features of the previous layer. The same idea has been extended

for deep neural networks [28, 29, 24, 30, 31, 26]. To disentangle intricate features, we can increase the depth of network by adding more hidden layers. Therefore, each layer extracts some features from its input and create their abstract representation. This concept of deep networks is similar to hierarchical neural networks like neocognitron model [32] but with some differences from architecture and learning algorithm. The neocognitron model uses winner-take-all unsupervised learning whereas learning in deep networks is done by back-propagation algorithm.

Deep learning networks have active applications in object recognition [33, 34, 35, 36], speech recognition [37, 38, 39], natural language processing [40, 41], theoretical science [42], medical science [43, 44, 45], etc. It is also being used commercially by companies like Google, Facebook, Twitter, Instagram, and others to provide better services to their customers.

There are various models of deep neural networks. One of which is deep auto-encoder. It is an extension of the auto-encoder neural network model. An auto-encoder network with multiple hidden layers is termed as deep auto-encoder [28]. It is generative model and extracts complex features with the help of multiple hidden layers. The learning of deep auto-encoder is not straightforward. As in case of conventional neural networks, performance of network is dependent on their weight initialization method. Poor weight initialization may results in network paralysis and network saturation states [16]. In addition to the above problem, the deep auto-encoders also encounter difficulty in propagating the gradients in backward direction [17].

To overcome the problem of weight initialization, Hinton et.al [28] introduced a different type of algorithm that learns one layer of features at a time. The procedure was termed by them as the "pretraining". This concept of pretraining use restricted Boltzmann machine (RBM) [46] as hidden units and pretrain the weights by contrastive divergence algorithm [47]. The RBMs are special kind of Boltzmann machine in which intra connections in a layer are not allowed. Here, RBMs are used as feature extractor to initialize the weights. During fine-tuning process, the units are replaced by sigmoid function and weight vectors are updated by back-propagation algorithm. With this two phase learning the network use both unsupervised and supervised learning. Supervised learning has some limitations like they required target labels to modify weights and also required some means by which they can communicate about error to lower level layers.

In [29], Hinton et. al. also proposed another generative model which used an unsupervised learning algorithm for fine-tuning of network. The authors proposed a deep belief network in

which top two layers having undirected connections form an associative memory and rest of the network has directed connections.

A variant of contrastive divergence (CD) is persistent CD (PCD) which is fast as compared to CD. In PCD [48], initialisation of Markov chain is done at the state in which it ends for the previous model and hence have a chance that it can jump from one valley to other. Different variants of these models have been proposed in literature as in [49, 45]. The paper [49] shows effect of depth on network's performance. These generative and unsupervised learning based models have some drawbacks like they can not use large image datasets because learning of such networks would be very time consuming. To solve this problem, either we can use supervised learning or may use same unsupervised learning with some additional approaches.

In the context of deep learning, one of the most prominent approach based on supervised learning is convolutional neural networks (CNNs). CNNs are based on Hubel and Wiesel's discovery of simple and complex cell [50]. They find their roots in neocognitron model proposed by Fukushima[32]. The convolutional networks create inbuilt invariance by using shared weights technique. Initially, CNNs with shared weight approach were applied to recognition of handwritten characters in [51]. In general, CNNs consist of three architectural ideas which are as follows: local receptive field, shared weights, and subsampling/pooling. The convolutional networks can be considered as analogues to conventional neural networks where each neuron is replaced by a feature map which is three dimensional. Instead of fully connected network as in conventional neural networks, patch of neurons is connected to one single neuron in a feature map of the next layer. The size of the patch is governed by the receptive field. Each convolutional layer is then followed by averaging/subsampling layer which makes representation robust with respect to noise and distortion in input.

Application of convolutional neural networks to pattern recognition has attracted researchers in machine learning community specifically in the field of computer vision. In [31], the LeNet-5 network with seven layers has been applied to MNIST data. The seven layers of the network are divided into two parts. First part consist of two convolutional layers and each layer is followed by the pooling layer and hence, first part has four layers. This part is responsible for extracting features from raw input. Second part is a multilayer perceptron network with one hidden layer which classify these features based on assumption that features are linearly separable.

This approach became more famous in 2012 after winning the ImageNet Large Scale Visual

Recognition Challenge (ILSVRC) [52] by AlexNet [33]. The network used the same approach as used by LeNet-5 but with more hidden layers (five convolutional layer and three fully connected layer). To reduce over-fitting, dropout technique has been used [53, 54]. After this, network with more hidden layers has become the foremost for feature extraction as done in [34, 55, 35, 36, 56]. It has proved its effectiveness in many applications. In 2014, Lin [57] proposed a different kind of convolutional architecture called 'network in network (NIN)' that used a small multilayer perceptron network in between convolutional layers. The idea was based on assumption that the features extracted by conventional CNNs are linearly separable, but this is not true in all cases. The representations that captured these concepts were generally highly non-linear function of the input. Since multilayer perceptron networks are universal function approximator, use of this can disentangle many nonlinear features easily. But what is effect of such approach on network's depth? Such type of approach has been used in [36] and may be these networks would be more valuable in near future.

The activation function of hidden units also effects network's performance. The LeNet-5 [31] network used hyperbolic tangent function. The activation function like sigmoidal and hyperbolic tangent work well with shallow neural networks [58]. But in case of deep neural networks, these units start saturating as error is propagated in backward direction via many hidden layers. Also, their weight matrices are very dense which creates more computational overhead for training of deep networks. The rectified linear units (ReLu) or rectified neurons are more better model of biological neurons [58]. The ReLu units has recently become a natural choice of practitioners for activation function.

Due to computational complexity of unsupervised learning based methods, they are unable to process large image datasets. To make them practically possible we can integrate generative models with convolutional models as done in [59, 45]. Use of convolutional and pooling operators make unsupervised learning based models more approachable specially in the field of medical science.

Deep learning achieves state-of-art performance in many applications specially in the field of computer vision. Still there are some restrictions which are limiting these networks in order to make them applicable for realistic learning task such as in robotics and online learning. Bengio et. al. in [60, 61] highlight these limitations and has discussed about major challenges of the field. These challenges include scalability, optimization, disentangling more explanatory features,

inference and sampling. Currently, optimization and scalability are conspicuous issues. Almost all models of deep neural networks used a kind of gradient descent algorithm which is more likely to get stuck in poor local minima, thereby degrading the performance of deep neural networks. Bengio in [62] introduced the idea of cultural and curriculum learning and suggest that the learning guided by intermediate terms is helpful for abstracting high-level features and also less likely to stuck in poor minima.

2.1 Optimization in Deep Learning Networks

Optimization is the process of searching the best possible solution to an optimization problem. Some of the well established optimization techniques in the literature include genetic algorithm, differential evolution, particle swarm optimization, ant colony optimization and many others [63, 64, 65, 66, 67, 68]. Each of these optimization techniques has their own set of operators and parameters.

An important and emerging research area in the field of deep learning networks is in optimization of the parameters of these networks. One of the known limitations associated with deep learning networks is that the architectures are usually heuristically designed. Choosing a suitable network for given problem is therefore a challenging task. Few studies have been reported in literature which incorporate hyper parameter and grid search optimization methods [69, 70]. Some of the recent studies raising the issue of optimization in deep neural networks include [71, 72, 73, 74, 75, 76]. Optimization of the deep learning network architectures, deep learning network parameters using evolutionary algorithm is still an emerging field as is evident from some of the research reported in [77, 78, 79, 80] which has a lot of scope to be explored further.

Recent applications of deep neural networks deal with large real life datasets like audio, visual, natural language processing and many others. Thus for solving such time consuming problems in real time, parallel implementation of the algorithm becomes unavoidable.

These computationally intensive algorithms combined with huge datasets for applications increase the need for greater computational speeds. The training process in deep neural networks is computationally very intensive. As the data size increases the process becomes more and more time consuming. Thus, arises the need for parallel implementation. In the context of parallel computation some of the parallel programming software tools used are Parallel Virtual Machine

(PVM) [81] and Message Passing Interface (MPI) [82] and GPU's for large scale neural networks training [83]. Some of the work involving the implementation of deep neural networks on GPU's has been reported in [83, 33]. The above mentioned software tools along with the associated parallel implementation strategies need further detailed analysis and can be helpful in tackling problems with large datasets in real time.

2.2 Applications

While dealing with images or spectral representation of spoken words, classical neural networks suffer from the problem of invariance. Deep convolutional neural networks solve problems by introducing invariance into the network structure itself. Thus they are good in handling recognition systems. The given dataset may have multiple instances of an image in different environment. With the aim of object classification or recognition, deep neural networks can perform the task efficiently with respect to translation and local distortion in given input. The well-known large datasets for image and speech are [84, 85, 86, 87, 88] and [89] respectively.

Unsupervised learning is more applicable to situations where we do not have sufficient labeled datasets. For example, in the field of medical science we would not have sufficient labeled datasets for a particular type of disease. In such circumstances deep neural networks based on unsupervised learning will be more efficient. We can apply these networks to solve different medical problems like diabetic retinopathy [43], detection of cancer cells in human body [90], spine imaging [91] and many others.

3 Framework of the Present Research

A brief review about the current trends in the deep learning paradigm has been discussed above and has identified wide vistas to explore. These include:

- investigations into various hybridized deep neural network systems;
- exploring their optimization and enhancement using different approaches of hybridization and learning like gradient descent, evolutionary strategies;
- applying such systems to problems with large datasets focussing on pattern recognition and classification.

It is proposed to limit our research in this thesis to an appropriate mixture and subset of the issues identified above.

References

[1] S. B. Lo, H. Chan, J. Lin, H. Li, M. T. Freedman, and S. K. Mun, "Artificial convolution neural network for medical image pattern recognition," *Neural Networks*, vol. 8, no. 7-8, pp. 1201–1214, 1995.

[2] C. M. Bishop, *Neural Networks for Pattern Recognition*. Oxford University Press, London, UK, 2000.

[3] J. Wood and J. Shawe-Taylo, "Invariant pattern recognition: A review," *Discrete Applied Mathematics*, vol. 69, no. 1-2, pp. 33–60, 1996.

[4] O. D. Trier, A. K. Jain, and T. Taxt, "Feature extraction methods for character recognition-a survey," *Pattern Recognition*, vol. 29, no. 4, pp. 641–662, 1996.

[5] G. P. Zhang, "Neural networks for classification: a survey," *IEEE Transactions on Systems, Man, and Cybernetics, Part C*, vol. 30, no. 4, pp. 451–462, 2000.

[6] A. Verikas and M. Bacauskiene, "Feature selection with neural networks," *Pattern Recognition Letters*, vol. 23, no. 1323–1335, 2002.

[7] K. Chen and L. A. Kurgan, "Neural networks in bioinformatics," in *Handbook of Natural Computing*, pp. 565–583, 2012.

[8] Y. Hayashi, J. J. Buckley, and E. Czogala, "Fuzzy neural network with fuzzy signals and weights," *Int. J. Intell. Syst.*, vol. 8, no. 4, pp. 527–537, 1993.

[9] R. P. Paiva and A. Dourado, "Interpretability and learning in neuro-fuzzy systems," *Fuzzy Sets and Systems*, vol. 147, no. 1, pp. 17–38, 2004.

[10] S. Mitra and Y. Hayashi, "Neuro-fuzzy rule generation: Survey in soft computing framework," *IEEE Trans. Neural Netw. Learning Syst.*, vol. 11, no. 3, pp. 748–768, 2000.

[11] A. Abraham and B. Nath, "Evolutionary design of neuro-fuzzy systems – a generic framework," in *In Proceedings of the 4th Japan- Australia Joint Workshop on Intelligent and Evolutionary Systems*, pp. 106–113, 2000.

[12] O. Cordón, "A historical review of evolutionary learning methods for mamdani-type fuzzy rule-based systems: Designing interpretable genetic fuzzy systems," *Int. J. Approx. Reasoning*, vol. 52, no. 6, pp. 894–913, 2011.

[13] D. Gardner, ed., *The Neurobiology of Neural Networks*. Computational neuroscience, Cambridge, Mass. MIT Press, 1993.

[14] M. A. Arbib, *The Handbook of Brain Theory and Neural Networks*. A Bradford Book, MIT Press, 1995.

[15] T. P. Trappenberg, *Fundamentals of Computational Neuroscience*. Oxford University Press, 2002.

[16] S. Kumar, *Neural Networks: A classroom approach*. Tata McGraw-Hill Education, 2004.

[17] S. Hochreiter, Y. Bengio, P. Fransconi, and J. Schmidhuber, "Gradient flow in recurrent nets: the difficulty of learning long-term dependencies," in *A Field Guide to Dynamical Recurrent Networks* (S. C. Kremer and J. F. Kolen, eds.), IEEE Press, 2001.

[18] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. Wiley, 2001.

[19] M. Dash and H. Liu, "Faeture selection for classification," *Intelligent Data Analysis*, vol. 1, pp. 131–156, 1997.

[20] S. Ledesma, G. Cerda-Villafana, G. Aviña, D. Hernández, and M. Torres, "Feature selection using artificial neural networks," in *MICAI 2008: Advances in Artificial Intelligence, 7th Mexican International Conference on Artificial Intelligence, Atizapán de Zaragoza, Mexico, October 27-31, 2008, Proceedings*, pp. 351–359, 2008.

[21] K. A. Glocer, D. Eads, and J. Theiler, "Online feature selection for pixel classification," in *Machine Learning, Proceedings of the Twenty-Second International Conference (ICML 2005), Bonn, Germany, August 7-11, 2005*, pp. 249–256, 2005.

[22] J. Wang, P. Zhao, S. C. H. Hoi, and R. Jin, "Online feature selection and its applications," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 3, pp. 698–710, 2014.

[23] Y. LeCun, Y. Bengio, and G. E. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, May 2015.

[24] G. E. Hinton, "Learning multiple layers of representation," *Trends in Cognitive Sciences*, vol. 11, no. 10, pp. 428–434, 2007.

[25] Y. LeCun, "Learning invariant feature hierarchies," in *Computer Vision - ECCV 2012. Workshops and Demonstrations - Florence, Italy, October 7-13, 2012, Proceedings, Part I* (A. Fusiello, V. Murino, and R. Cucchiara, eds.), pp. 496–505, 2012.

[26] G. E. Hinton, "Where do features come from?," *Cognitive Science*, vol. 38, pp. 1078–1101, August 2014.

[27] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Networks*, vol. 61, pp. 85–117, 2015.

[28] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, pp. 504–507, July 2006.

[29] G. E. Hinton, S. Osindero, and Y. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, pp. 1527–1554, July 2006.

[30] Y. LeCun and Y. Bengio, "Convolutional networks for images, speech, and time-series," in *The Handbook of Brain Theory and Neural Networks* (M. A. Arbib, ed.), pp. 255–257, MIT Press, 1995.

[31] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," in *Intelligent Signal Processing* (S. Haykin and B. Kosko, eds.), vol. 86, pp. 306–351, IEEE Press, 2001.

[32] K. Fukushima and S. Miyake, "Neocognitron: A new algorithm for pattern recognition tolerant of deformations and shifts in position," *Pattern Recognition*, vol. 15, no. 6, pp. 455–469, 1982.

[33] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held on December 3-6, 2012*, (Lake Tahoe, Nevada, United States), pp. 1106–1114, 2012.

[34] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Computer Vision - ECCV 2014 - 13 European Conference*, (Zurich, Switzerland), pp. 818–833, September 6-12 2014.

[35] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-sacle image recognition," *Computing Research Repository (CoRR)*, vol. abs/1409.1556, 2014.

[36] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," *Computing Research Repository (CoRR)*, vol. abs/1409.4842, 2014.

[37] G. E. Hinton, L. Deng, D. Yu, G. E. Dahl, A. R. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, and B. Kingushy, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE, Signal Process. Mag.*, vol. 29, no. 6, pp. 82–97, 2012.

[38] O. Abdel-Hamid, A. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu, "Convolutional neural networks for speech recognition," *IEEE/ACM Transactions on Audio, Speech & Language Processing*, vol. 22, no. 10, pp. 1533–1545, 2014.

[39] T. N. Sainath, B. Kingsbury, G. Saon, H. Soltau, A. Mohamed, G. E. Dahl, and B. Ramabhadran, "Deep convolutional neural networks for large-scale speech tasks," *Neural Networks*, vol. 64, pp. 39–48, 2015.

[40] R. Socher, Y. Bengio, and C. D. Manning, "Deep learning for NLP (without magic)," in *The 50th Annual Meeting of the Association for Computational Linguistics, Tutorial Abstracts, July 8, 2012, Jeju Island, Korea*, vol. 5, 2012.

[41] R. Sarikaya, G. E. Hinton, and A. Deoras, "Application of deep belief networks for natural language understanding," *IEEE/ACM Transactions on Audio, Speech & Language Processing*, vol. 22, no. 4, pp. 778–784, 2014.

[42] "Kaggle: Higgs boson machine learning challenge." http://www.kaggle.com/c/higgs-boson, September 2014.

[43] "Kaggle: Diabetic rentinopathy detection challenge." https://www.kaggle.com/c/diabetic-retinopathy-detection, July 2015.

[44] A. D. Brebisson and G. Montana, "Deep neural networks for anatomical brain segmentation," *Computing Research Repository (CoRR)*, vol. abs/1502.02445, February 2015.

[45] H. Shin, M. Orton, D. J. Collins, S. J. Doran, and M. O. Leach, "Stacked autoencoders for unsupervised feature learning and multiple organ detection in a pilot study using 4d patient data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1930–1943, 2013.

[46] D. H. Ackley, G. E. Hinton, and T. J. Sejnowski, "A learning algorithm for boltzmann machines," *Cognitive Science*, vol. 9, pp. 147–169, January 1985.

[47] M. A. Carreira-Perpinan and G. E. Hinton, "On contrastive divergence learning," in *Proceeding of the 10th International Workshop on Artificial Intelligence and Statistics (AISTATS)*, (Bridgetown, Barbados), January 6-8 2005.

[48] T. Tieleman, "Training of restricted botzmann machines using approximations to the likelihood gradient," in *Machine Learning, Proceddings of the Twenty-Fifth Internaltional Conference (ICML)*, (Helsinki, Finland), pp. 1064–1071, June 5-9 2008.

[49] Y. Wu and H. J. Cai, "A simulation study of deep belief network combined with the self-organizing meachnism of adaptive resonance theory," in *International conference on computational intelligence and software engineering (CiSE)*, (Wuhan), pp. 1–4, December 10-12 2010.

[50] D. Hubel and T. Wiesel, "Receptive fields, binocular interaction, and functional architecture in the cat's visual cortex," *Journal of Physiology*, vol. 160, pp. 106–154, 1962.

[51] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropogation applied to handwritten zip code recognition," *Neural Computation*, vol. 1, no. 4, pp. 541–551, 1989.

[52] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision (IJCV)*, pp. 1–42, April 2015.

[53] G. E. Dahl, T. N. Sainath, and G. E. Hinton, "Improving deep neural networks for LVCSR using rectified linear units and dropout," in *IEEE International Conference on Acoustics,*

Speech and Signal Processing, (ICASSP), (Vancouver, British Columbia, Canada), pp. 8609–8613, May 26-31 2013.

[54] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Droupout: A simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[55] I. J. Goodfellow, Y. Bulatov, J. Ibraz, S. Arnoud, and V. Shet, "Multi-digit number recognition from street view imagery using deep convolutional neural networks," *Computing Research Repository (CoRR)*, vol. abs/1312.6082, 2013.

[56] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, "Striving for simplicity: The all convolutional net," *Computing Research Repository (CoRR)*, vol. 1412.6806, April 2015.

[57] M. Lin, Q. Chen, and S. Yan, "Network in network," *Computing Research Repository (CoRR)*, vol. abs/1312.4400, 2013.

[58] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proceeding of the 14th International Conference on Artificial Intelligence and Statistics (AISTATS - 11)*, pp. 315–323, 2011.

[59] H. Lee, R. B. Grosse, R. Ranganath, and A. Y. Ng, "Unsupervised learning of hierarchical representations with convolutional deep belief networks," *Commun. of ACM*, vol. 54, no. 10, pp. 95–103, 2011.

[60] Y. Bengio, A. C. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, 2013.

[61] Ç. Gülçehre and Y. Bengio, "Knowledge matters: Importance of prior information for optimization," *Computing Research Repository (CoRR)*, vol. abs/1301.4083, 2013.

[62] Y. Bengio, "Evolving culture versus local minima," in *Growing Adaptive Machines - Combining Development and Learning in Artificial Neural Networks*, pp. 109–138, Springer, 2014.

[63] J. F. Frenzel, "Genetic algorithm," *IEEE Potentials*, vol. 12, no. 3, pp. 21–24, 1993.

[64] S. Das and P. N. Suganthan, "Differential evolution: A survey of the state-of-the-art," *IEEE Trans. Evolutionary Computation*, vol. 15, no. 1, pp. 4–31, 2011.

[65] R. Poli, J. Kennedy, and T. Blackwell, "Particle swarm optimization," *Swarm Intelligence*, vol. 1, no. 1, pp. 33–57, 2007.

[66] M. Dorigo and K. Socha, "Ant colony optimization," in *Handbook of Approximation Algorithms and Metaheuristics*, 2007.

[67] M. Dorigo and T. Stützle, *Ant Colony Optimization*. MIT Press, 2004.

[68] M. Dorigo and C. Blum, "Ant colony optimization theory: A survey," *Theor. Comput. Sci.*, vol. 344, no. 2-3, pp. 243–278, 2005.

[69] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *Journal of Machine Learning Research*, vol. 13, pp. 281–305, 2012.

[70] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2010, Chia Laguna Resort, Sardinia Italy, May 13-15, 2010*, pp. 249–256, 2010.

[71] M. Kim and L. Rigazio, "Deep clustered convolutional kernels," *Computing Research Repository (CoRR)*, vol. abs/1503.01824, 2015.

[72] T. N. Sainath, B. Kingsbury, H. Soltau, and B. Ramabhadran, "Optimization techniques to improve training speed of deep neural networks for large speech tasks," *IEEE Transactions on Audio, Speech & Language Processing*, vol. 21, no. 11, pp. 2267–2276, 2013.

[73] J. Martens, "Deep learning via hessian-free optimization," in *Proceeding of the 27th International Conference on Machine Learning (ICML-10)*, (Haifa, Israel), pp. 735–742, Jun 21-24 2010.

[74] D. C. Ciresan, U. Meier, and J. Schmidhuber, "Multi-column deep neural networks for image classification," in *2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, June 16-21, 2012*, pp. 3642–3649, 2012.

[75] Q. V. Le, J. Nigam, A. Coates, A. Lahiri, B. Prochnow, and A. Y. Ng, "On optimization methods for deep learning," in *Proceedings of the 28th International Conference on Machine Learning ICML 2011, Bellevue, Washington, USA, June 28-July 2, 2011*, pp. 265–272, 2011.

[76] S. Anwar, K. Hwang, and W. Sung, "Fixed point optimization of deep convolutional neural networks for object recognition," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2015, South Brisbane, Queensland, Australia, April 19-24, 2015*, pp. 1131–1135, 2015.

[77] O. E. David and I. Greental, "Genetic algorithms for evolving deep neural networks," in *GECCO Comp'14 Proceedings of the 2014 conference companion on genetic and evolutionary computation companion*, pp. 1451–1452, 2014.

[78] J. D. Lamos-Sweeney, "Deep learning using genetic algorithm," Master's thesis, Rochester Institute of Technology, Rochester, NY, May 2012.

[79] B. Cheung and C. Sable, "Hybrid evolution of convolutional networks," in *10th International Conference on Machine Learning and Applications and Workshops (ICMLA) 2011, Honolulu, HI, December 10-21.*, vol. 1, pp. 293–297, IEEE, 2011.

[80] T. H. Maul, A. Bargiela, C. S. Yew, and A. S. Adamu, "Towards evolutionary deep neural networks," in *Proceedings 28th European Conference on Modelling and Simulation*, 2014.

[81] A. Geist, A. Beguelin, J. Dongarra, W. Jiang, R. Manchek, and V. Sunderam, *PVM: Parallel Virtual Machine—A Users' Guide and Tutorial for Networked Parallel Computing.* MIT Press, 1994.

[82] M. J. Quinn, *Parallel Programming in C with MPI and OpenMP.* Tata McGraw-Hill Higher Education, 2003.

[83] A. Krizhevsky, "One weird trick for parallelzing convolutional neural networks," *Computing Research Repository (CoRR)*, vol. abs/1404.5997, 2014.

[84] Y. LeCun, C. Cortes, and C. J. C. Burges, "The mnist database for handwritten digits." http://yann.lecun.com/exdb/mnist/.

[85] F. J. Huang and Y. LeCun, "The norb dataset." http://www.cs.nyu.edu/ ylclab/data/norb-v1.0/, July 2004.

[86] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples an incremental bayesian approach tested on 101 object categories," in *Proceedings of the Workshop on Generative-Model Based Vision*, (Washington, DC), June 2004.

[87] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *In CVPR*, 2009.

[88] A. Krizhevsky, "Learning multiple layers of features from tiny images," tech. rep., 2009.

[89] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, and N. L. Dahlgren, "Darpa timit acoustic phonetic continuous speech corpus cdrom," 1993.

[90] "The cancer genome atlas." http://www.cancerimagingarchive.net/.

[91] "Spineweb: Collaborative platform for research on spine imaging and image analysis." http://spineweb.digitalimaginggroup.ca/.